ECE 425 Introductions to Microprocessors
Laboratory Work 9

Objective:

1) Software Interrupts.
2) Timers and Counters, Timer Interrupt.
3) Interrupt Simulation Using Stimulus.

Preparation:

Option register and interrupt control register are used to program interrupt resources and their register bits are shown in the figure below.

OPTION REGISTER

RBPU	INTEDG	TOCS	TOSE	PSA	PS2	PS1	PS0

INTCON REGISTER

GIE	EEIE	TOIE	INTE	RBIE	TOIF	INTF	RBIF

1) In the program below timer interrupt is enabled and as clock course RA4/TOCKI is used. Send pulses to RA4/TOCKI using stimulus. Simulate the program using F7 button and stimulus, comment on the results.

```
LIST P=16F84A
INCLUDE "P16f84A.INC"

__config _CP_OFF&_WDT_OFF&_XT_OSC
        org 0x00;
        goto start
        org 0x04;
        goto ISR_TOCKI;
start
        bsf     STATUS, RP0;
        movlw   0xFF;
        movwf   TRISA;
        bsf     OPTION_REG, T0CS;
        bsf     OPTION_REG, T0SE;
        bcf     STATUS, RP0;
        clrf    PORTB;

        bsf     INTCON, GIE;
        bsf     INTCON, T0IE; Enable Timer/Counter Interrupt
        bcf     INTCON, T0IF;
loop
        nop;
        nop;
        nop;
        nop;
goto loop;

 ISR_TOCKI
        bcf     INTCON, T0IF;
        nop;
        nop;
        nop;
        retfie;
  end
```

2) Using stimulus trace the following program. Send pulses to RA4/TOCKI. What happens when the timer overflows? Simulate the program using stimulus.

```
LIST P=16F84A
INCLUDE "P16f84A.INC"

__config _CP_OFF&_WDT_OFF&_XT_OSC

        org 0x00;
        goto start
        org 0x04;
        goto ISR_TOCKI;

start
        bsf     STATUS, RP0;
        movlw   0xFF;
        movwf   TRISA;
        bsf     OPTION_REG, T0CS;
        bsf     OPTION_REG, T0SE;

        bcf     STATUS, RP0;
        clrf    PORTB;

        bsf     INTCON, GIE;
        bsf     INTCON, T0IE;
        bcf     INTCON, T0IF;

        movlw .253;
        movwf TMR0;

loop
        nop;
        nop;
        nop;
        nop;
goto loop;

 ISR_TOCKI

        bcf     INTCON, T0IF;
        nop;
        nop;
        nop;
        retfie;
end
```

3) In the following assembly program, timer value is incremented at each clock cycle. Simulate the program using F7 and comment on the results.

```
LIST P=16F84A
INCLUDE "P16f84A.INC"

__config _CP_OFF&_WDT_OFF&_XT_OSC

            org 0x00;
            goto start
            org 0x04;
            goto ISR_TOCKI;

start
            clrf TMR0;

            bsf         STATUS, RP0;
            bcf         OPTION_REG, T0CS; At each clock cycle, timer value will be incremented
            ;bsf        OPTION_REG, PSA; Prescalar value is assigned to WDT by default

            bcf         STATUS, RP0;

            bsf         INTCON, GIE;
            bsf         INTCON, T0IE;
            bcf         INTCON, T0IF;

loop
            nop;
            nop;
            nop;
            nop;

goto loop;

 ISR_TOCKI

            bcf         INTCON, T0IF;
            nop;
            nop;
            nop;

            retfie;
end
```

4) In the following assembly program, timer value is incremented at each clock cycle. Simulate the program using F7 and comment what happens when timer overflows?

```
LIST P=16F84A
INCLUDE "P16f84A.INC"

__config _CP_OFF&_WDT_OFF&_XT_OSC

            org 0x00;
            goto start
            org 0x04;
            goto ISR_TOCKI;

start
            movlw .240;
            movwf TMR0;

            bsf         STATUS, RP0;
            bcf         OPTION_REG, T0CS; At each clock cycle, timer value will be incremented


            bcf         STATUS, RP0;
            clrf        PORTB;

            bsf         INTCON, GIE;
            bsf         INTCON, T0IE;
            bcf         INTCON, T0IF;

loop
            nop;
            nop;
            nop;
            nop;
goto loop;

 ISR_TOCKI

            bcf         INTCON, T0IF;
            nop;
            nop;
            nop;

            retfie;

end
```

5) In the following assembly program, timer value is incremented at every two clock cycle. Simulate the program using F7 and comment what happens when timer overflows?

```
LIST P=16F84A
INCLUDE "P16f84A.INC"

__config _CP_OFF&_WDT_OFF&_XT_OSC

        org 0x00;
        goto start
        org 0x04;
        goto ISR_TOCKI;

start
        movlw .240;
        movwf TMR0;

        bsf     STATUS, RP0;

        bcf     OPTION_REG, T0CS; At each clock cycle, timer value will be incremented
        bcf     OPTION_REG, PSA; Prescalar value is assigned to TMR0 register
        bcf     OPTION_REG, PS2;
        bcf     OPTION_REG, PS1;
        bcf     OPTION_REG, PS0;
        ; TMR0 rate is 1:2 which means that every 2 clock cycles timer value will be incremented
        ; For the other prescalar values see the lecture notes

        bcf     STATUS, RP0;

        bsf     INTCON, GIE;
        bsf     INTCON, T0IE;
        bcf     INTCON, T0IF;

loop
        nop;
        nop;
        nop;
        nop;
goto loop;

 ISR_TOCKI

        bcf     INTCON, T0IF;
        nop;
        nop;
        nop;

        retfie;

end
```

Laboratory Work:

1) Trace program segments in preparation 1-5 and comment on the results.
2) Program your timer register such that it generates interrupt signal every 65.5msec.

During your LAB work show every step that you complete to the LAB assistant. Get a copy of assembly files you write during the LAB hour via a flash disk for future reference.